(12) UK Patent Application (19) GB (11) 2 180 725 (13) A

(43) Application published 1 Apr 1987

(21) Application No 8523183

(22) Date of filing 19 Sep 1985

(51) INT CL[4]
H03M 5/12 H04L 7/02

(52) Domestic classification (Edition I)
H4P DBB

(56) Documents cited
None

(58) Field of search
H4P
Selected US specifications from IPC sub-class H04L

(71) Applicant
STC plc,

(Incorporated in United Kingdom),

190 Strand, London WC2R 1DU

(72) Inventors
David Crawford Odhams,
Geoffrey Brian Donald Brown

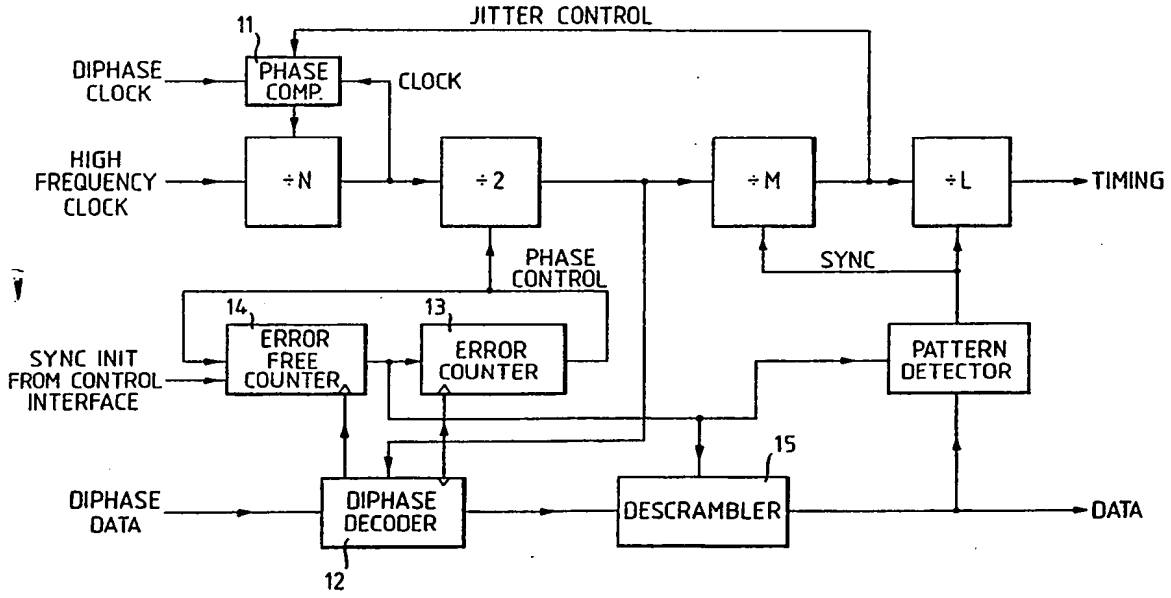
(74) Agent and/or Address for Service
S. R. Capsey, STC Patents, Edinburgh Way, Harlow, Essex CM20 2SH

(54) Digital communication circuit

(57) A digital circuit which forms the main element of a line interface, between an exchange and the line or between the line and a subscriber's terminal is moduler in structure. The intelligence being conveyed includes speech in digital form. One of the modules includes error-checking, based on the line code used, in which binary 1 is sent as 1–0 and binary 0 as 0–1. Hence 00 and 11 are incorrect. The checking involves counting the incorrect and correct codes as received, an incorrect code usually occurring as a result of a spurious phase-shift. If two incorrect codes occur before or at the same time as up to 16 correct ones, an error is assumed and the data extraction stopped. In addition the clock phase is inverted, which if the error is due to phase shift usually cures the thing. If 16 correct codes occur *before* 2 wrong ones, correct operation is assumed and data extraction continued.

Note that in some cases the number of correct and incorrect codes would differ from those just mentioned.

Fig.3.



The drawing(s) originally filed was (were) informal and the print here reproduced is taken from a later filed formal copy.

GB 2 180 725 A

Fig. 1.

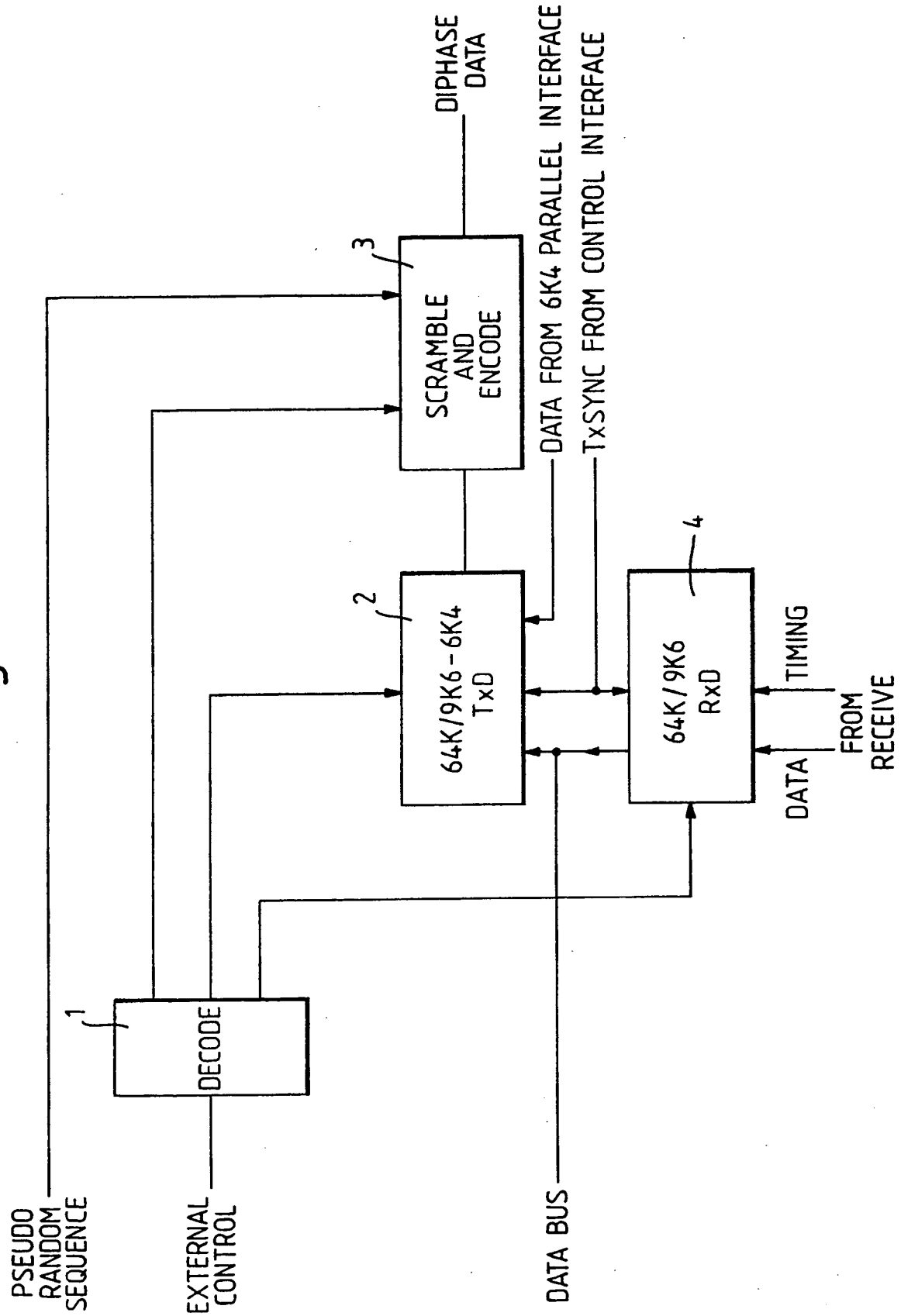


Fig.2.

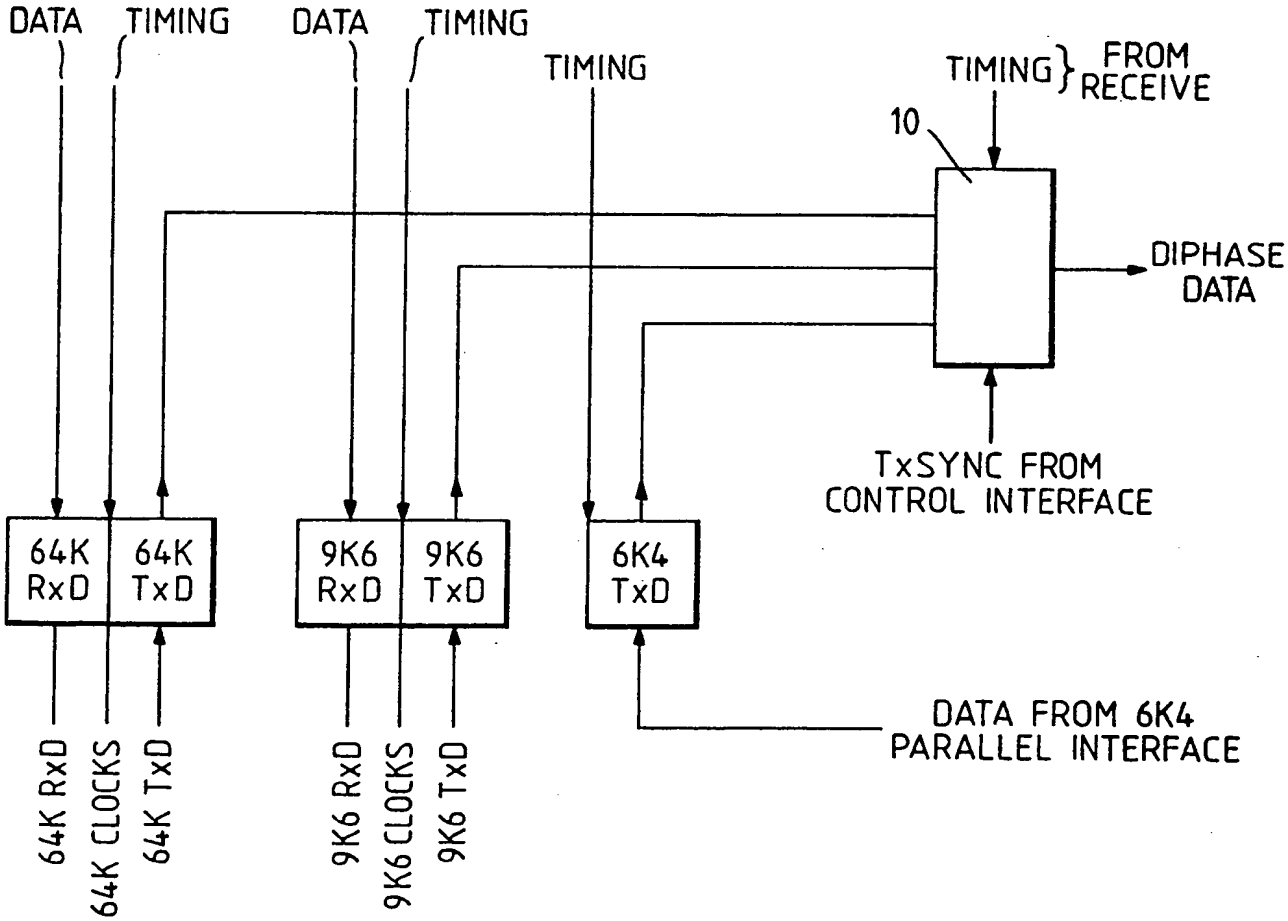
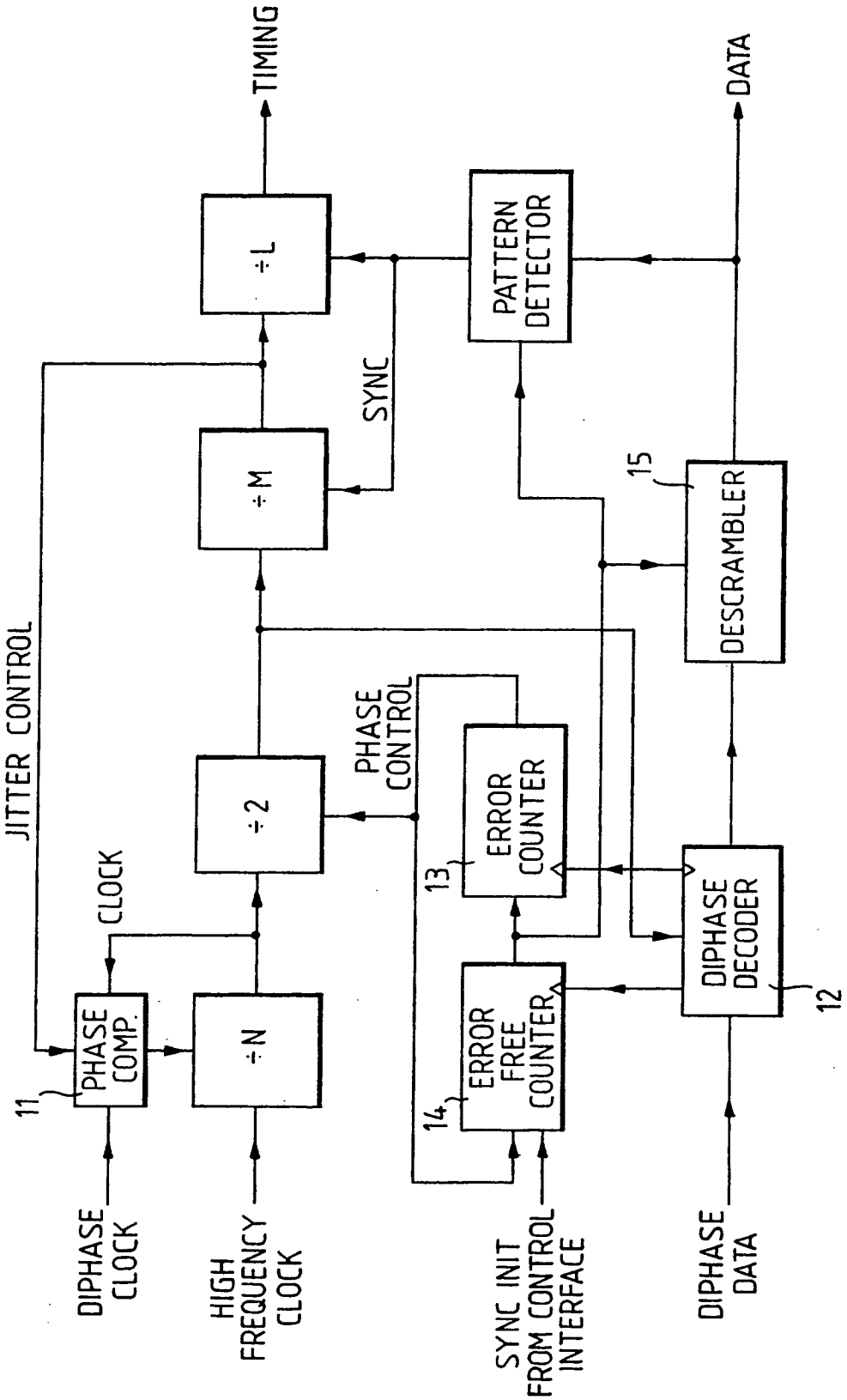


Fig.3.



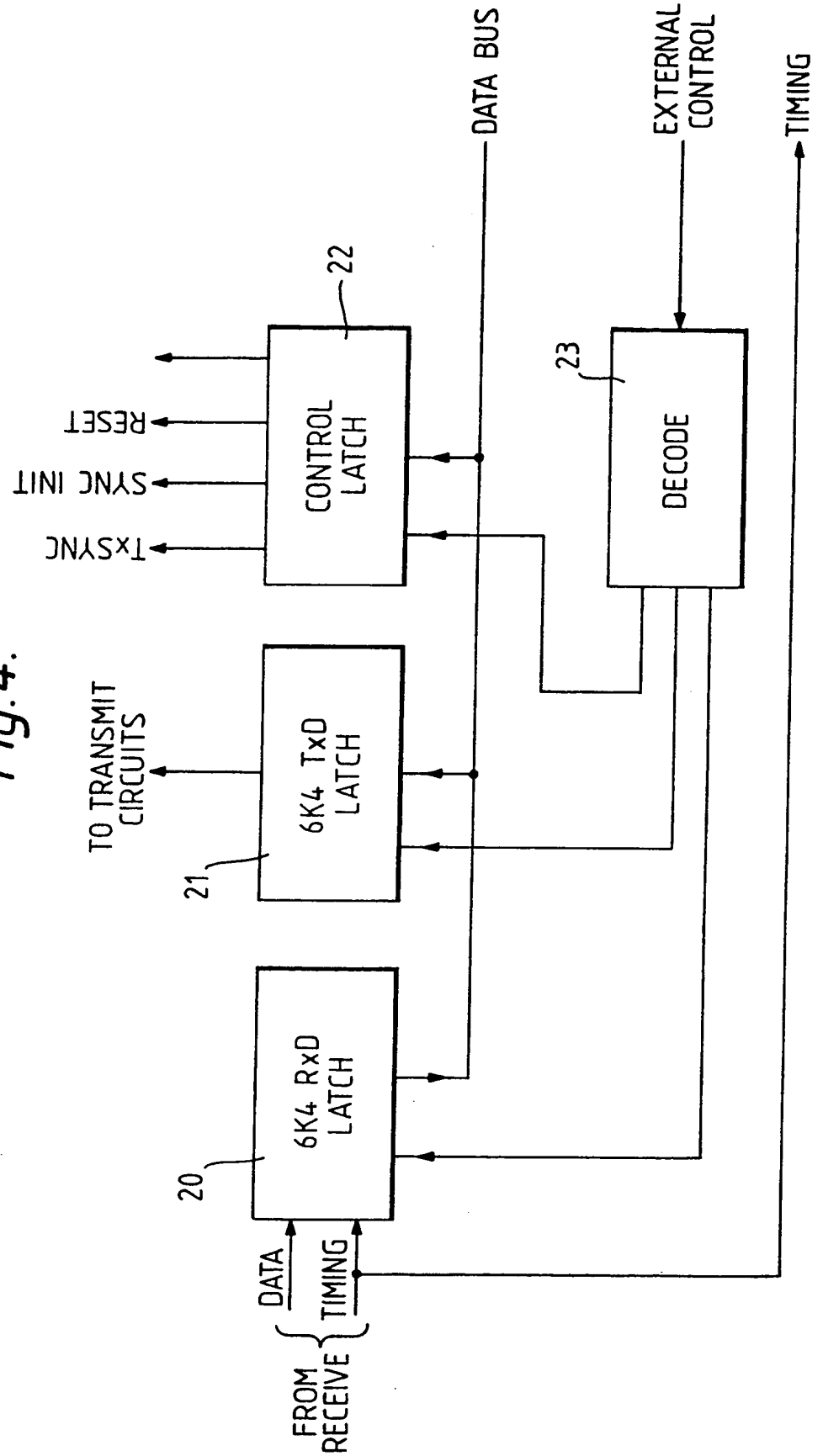


Fig.4.

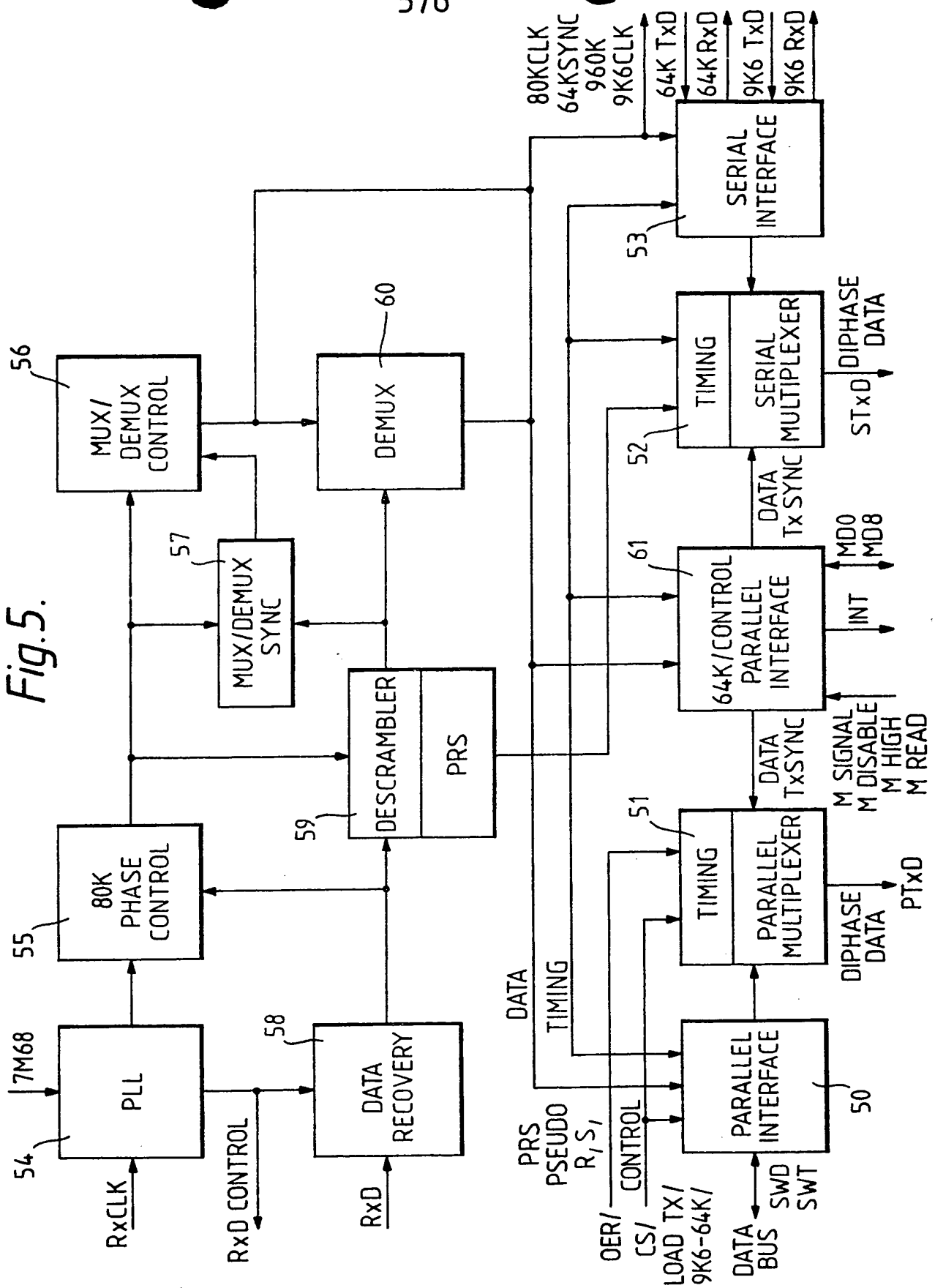


Fig.5.

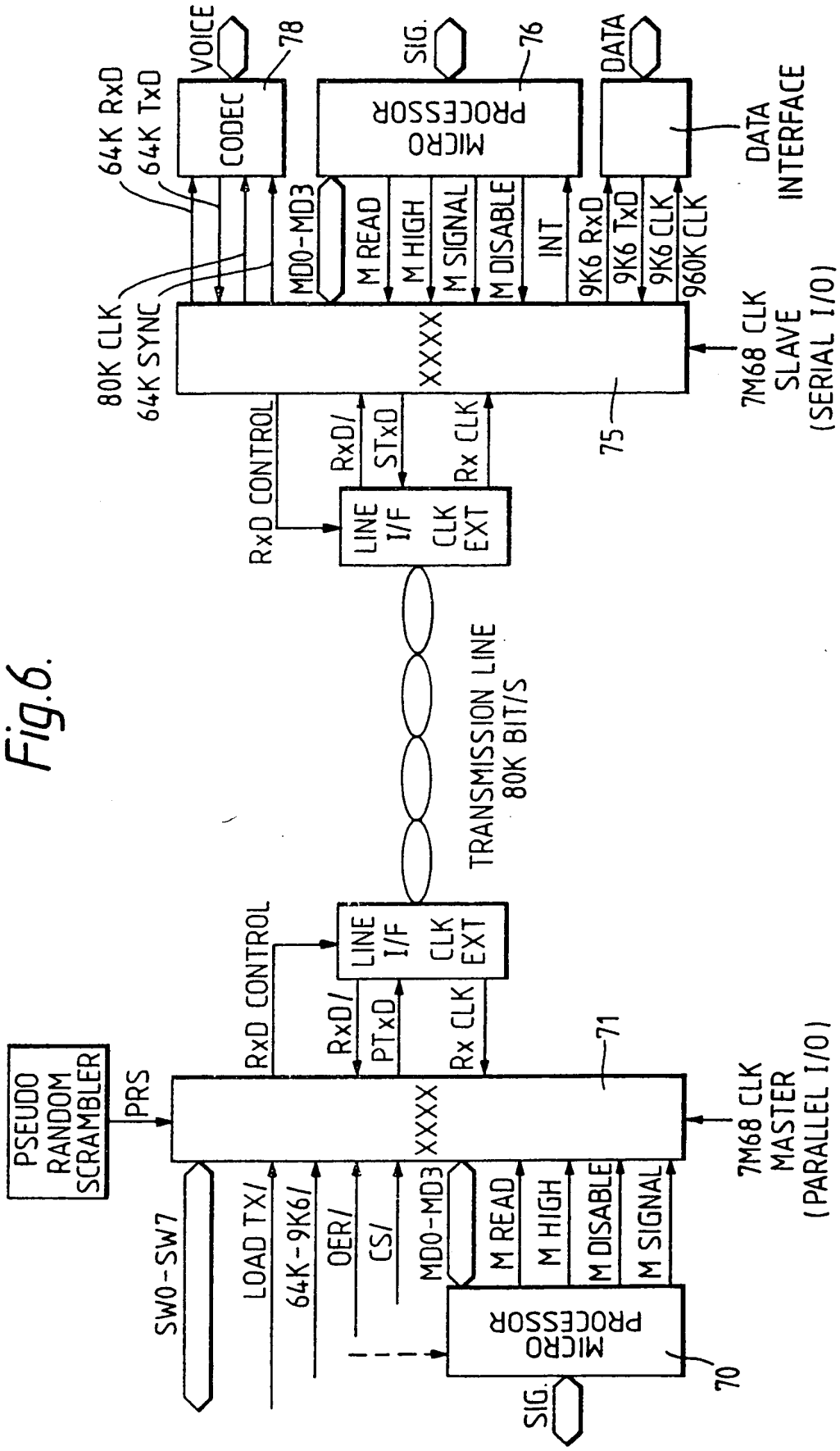


Fig.6.

SPECIFICATION

Digital communication circuit

This invention relates to a digital communication circuit, for use, for example, as a link between a PABX and the equipment at a subscriber's premises.

The arrangement to be described conveys three independent-digital channels for transmission over a bidirectional serial data link. It operates as a master-slave pair with one device at each end of the transmission link. The serial data sequence is scrambled with a pseudo-random sequence to obtain a balanced data stream, which encoded in the WAL1 diphase format. This latter is a code in which one binary value is sent as a square (or as nearly square as line conditions allow) wave; with, for example binary 1 sent as 1-0 and binary 0 as 0-1.

The three channels are at bit rates 64K, 9K6 (i.e. 9,600 bits/sec.) and 6K4 bits/sec.

The serial data stream as transmitted is in frames, each divided into two sub-frames, with each sub-frame defined as five words, each of ten bits. A frame is the smallest division of the serial bit stream to accommodate one eight-bit byte of the 6K4 channel. Each sub-frame is the smallest division of the serial bit stream to accommodate an integer number of bits of the 9K6 channel.

Each word is the smallest division of the serial bit stream to accommodate one eight-bit byte of the 64K channel. This is shown in the table :

TRANSMITTED SEQUENCE

A	B	WORD 0–)	
A	C	WORD 1–)	SUB-
A	B	WORD 2–)	FRAME
A	C	WORD 3–)	0
A	B	WORD 4–)	
A	B	WORD 5–)	
A	C	WORD 6–)	SUB-
A	B	WORD 7–)	FRAME
A	C	WORD 8–)	1
A	B	WORD 9–)	

Here A represents eight bits of the 64K channel (speech in the PABX example), B represents 2 bits of the 9K6 channel and C represents two bits of the 6K4 channel.

This bit allocation gives the optimum arrangement consistent with (a) demultiplexing the 64K channel to a standard CODEC, (b) minimum bit storage to adapt the 9K6 channel to a regular bit period, and (c) sufficient consecutive bits in the 6K4 channel to enable simple synchronisation. The resulting serial bit stream is scrambled with a pseudo-random bit sequence, and then diphase encoded.

An object of the invention is to provide a system in which intelligence can be transmitted in an economical manner, with an error checking facility.

According to the invention, there is provided an electronic circuit unit for data handling, the data being expressed in a diphase code, wherein one binary value is represented by 01 and the other binary value by 10, so that 01 and 10 are correct bit conditions and 00 and 11 are incorrect bit conditions, in which if *a* such incorrect bit conditions are counted by a first counter before or at the same time as *b* such correct conditions are counted by a second counter it is assumed that an error condition exists, in which outputs from the counters cause inversion of the phase of the output from a clock circuit used in association with the data recovery means when such an error condition is detected, the output of such data being interrupted when such an error condition is detected, in which if the second counter counts *b* successive correct bit conditions before the first counter counts *a* incorrect conditions the data output occurs, and in which on each occasion that the counters detect an error and also each occasion that the second counter reaches *b* the counters are cleared.

An embodiment of the invention will now be described with reference to the accompanying drawings, in which :

Figure 1 is a master transmit and 64K/9K6 parallel interface.
Figure 2 is a slave transmit and 64K/9K6 serial interface.
Figure 3 is a receive module.
Figure 4 is a 6K4 parallel interface and control latches assembly.
Figure 5 is an equipment assembly embodying the invention and using the module of Figures 1 to 4.
Figure 6 is a simplified schematic of an application of assemblies such as that of Figure 5.

Master transmit and 64K/9K6 parallel interface (figure 1)

External control signals for this module are decoded in a decoder 1, to load the 64K channel eight-bit bytes and 9K6 channel six-bit bytes into a set of shift registers 2 from an external data bus. Eight-bit bytes for the 6L4 channels are also loaded from the 6K4 parallel interface (Figure 4).

The multiplexing is achieved by interleaving the lowest four bits of the 6K4 channel byte with the first byte of

the 9K6 channel. The ten bits thus assembled are then added two at a time to the first five bytes of the 64K channel. This process is then repeated for the highest four bits of the 6K4 channel byte, giving the frame structure shown in the Table, above.

The serial bit stream is then passed from the block 2 to the Scramble and Encode block 3, where it is EXCLUSIVE OR'ed with a pseudo-random sequence of bits. The combined data streams are then diphase encoded and transmitted to the other end of the line.

Received 64K channel bytes, and 9K6 channel bytes are demultiplexed by the Receive module Figure 3, and loaded into latches 4, where they are read in response to external control signals.

Slave transmit and 64K/9K6 serial interface (figure 2)

Control signals from the receive module (Figure 3) provide the control for external logic to send from and units to the 64K and 9K6 channels serially. The 64K channels thus obtained are suitable for driving a CODEC directly. The 9K6 channel signals adjust the 9K6 data to a constant period bit sequence. The module also reads 6K4 data from the 6K4 parallel interface (Figure 4). The mulitplexer 10 in this module is controlled by the receive module, and combines the three channels for transmission as described above.

The serial bit stream in then EXCLUSIVE OR'ed with a pseudo-random sequence of bits, whereafter the combined data streams are diphase encoded and transmitted to the other end of the link.

This interface is at the opposite end of the line from the one described above. An interface such as shown in Figure 4 and a receive module are also provided at the same end of the line as the interface of Figure 2.

Self synchronising receive module (figure 3)

The dividers (–N, –2 and –M) and the phase comparator 11 form a digital phase locked loop, locking the output of the –N divider with the incoming clock signal. To synchronise the data, the remainder of the circuit is initialised with the signal line "SYNC INIT'. The diphase decoder 12, error counter 13 and error free counter 14 ensure that the diphase data is being correctly decoded. This is done by adjusting the phase of the bit clock via the control connection PHASE CONTROL until the two halves of the diphase data differ for a minimum period determined by the error free counter 14 whilst allowing a percentage of transmission errors determined by the error counter 13. When the error free counter overflows, the phase of the bit clock is locked and the error counter is disabled. When the error counter 13 overflows both counters are reset and the phase of the bit clock inverted.

Whilst the error free counter 14 is running the descrambler 15 reads the incoming data. When the error free counter 14 overflows the descrambler 15 regenerates the pseudo-random sequence to descramble the data. The diphase data has to be the pure pseudo-random sequence at this time, and the length of the error free counter 14 is sufficient for the descrambler to regenerate the pseudo-random sequence correctly.

When the descrambler 15 begins to decode the diphase data the pattern detector searches for a knon-non-zero bit pattern with which to initialise the –M and –L dividers as channel demultiplexers.

Demultiplexed data is then directed to the serial 9K6 and 64K module, the parallel 9K6 and 64K module and the 6K4 parallel module.

We now consider the above synchronisation process in more detail. It will be recalled that the serial bit stream with which the receiver is presented has transitions at (i) all centre cell locations, and (ii) cell boundary locations when two consecutive data bit values are the same logic state 1 or 0. If the data is all logic 1's or all logic 0's, the transitions occur at twice the data rate, and the receiver is unable to tell one from the other. This explains the need for a high degree of synchronisation.

The phase lock loop referred to above locks on to the transitions, and generates a clock at twice the data rate. This clock can be divided by two to generate a clock at the frequency of the data, but as yet the phase of the clock is unknown.

By inspecting each pair of encoded data bits received, i.e. the received data in the two halves of the data clock period, the following four codes can be received.

(a) 00 An undecodable sequence; either a transmission error, or the second cell of a logic 1 followed by the first cell of a logic 0, i.e. the data clock is 180° out of phase.

(b) 01 This is the coding sequence for a logic 0 if the clock is in phase, or the second cell of a logic 1 followed by the first cell of a logic 1 if the data clock is 180° out of phase.

(c) 10 This is the coding sequence for a logic 1 if the clock is in phase, or the second cell of a logic 1 followed by the first cell of a logic 1 if the clock is 180° out of phase.

(d) 11 Another undecodable sequence; either a transmission error, or the second cell of a logic 0 followed by the first cell of a logic 1 if the clock is 180° out of phase.

The data scrambler, which is part of the scramble and encode block 3 of Figure 1 has as one of its functions ensuring that there is a limited number of consecutive logic 1's and 0's. This ensures that the bit sequences 00 and 11, between two cells, occurs frequently. This does not cause decoding difficulties if the clock which controls decoding is properly in phase with the bit stream being decoded.

We now consider the error counter 13 and the error free counter 14 in more detail. The error counter counts the occurences of either 00 or 11 codes within a data cell, and the error free counter counts 01 or 10 codes within a data cell.

These counters are cleared by an external command when a data stream is to be handled, and commences

to count. When the error free counter 14 is cleared it switches the descrambler 15 from "run" to "load". If the error counter 13 overflows, i.e. reaches a count of 2, either before or as the error free counter 14 overflows, i.e. reaches a count of 16, three things occur:

(i) the divider generating the data clock from the bit clock is forced to invert the phase of the data clock;

(ii) the error free counter 14 is cleared;

(iii) the error counter 13 is cleared;

If the error free counter 14 overflows *before* the error counter 13, then :

(i) the error counter 13 is hold cleared;

(ii) the descrambler 15 is switched from "load" to "run".

The control of the descrambler 15 assumes that the data being received is the encoded sequence of the output of the transmission end's scrambler in block 3, Figure 1. The descrambler 15 is a seven-stage shift register with EXCLUSIVE-NOR feedback.

When the descrambler 15 is switched to "load" it clocks the received data directly into its seven-stage shift register. When it is switched to "run" it clocks in the EXCLUSIVE-NOR feedback. Thus the minimum number of bits required to be clocked is seven.

6K4 Parallel interface and control latches (figure 4)

The 6K4 parallel interface latches comprise :

(a) The received data latch 20, loaded by the receive module (Figure 3) and read by the external logic.

(b) The transmit data latch 21, loaded by the external logic and read by the master and slave transmit modules.

(c) The control latch 22, loaded by the external logic and providing control signals to the other modules.

(d) BUS Control, included in the control latch block 22, which determines whether the internal latch reads from or writes to the data bus.

All three latches use a common data bus. The external control signals are decoded in a decode block 23.

We now consider Figures 4 and 1 together. The clock to the M block is at 80 KHz, and the –M block divides by ten. This enables the A channel, 64 Khz, to be separated from the B and C channels, and the –L clock, which is clocked at 8 KHz, divides by five to enable the B channel to be separated from the C channel.

The equipment assembly (Figure 5)

Figure 5 is a block diagram of an integrated circuit chip embodying the circuitry shown in Figures 1, 2, 3 and 4, and one of these chips would be used at each end of the line with different blocks effectively used, depending on whether the chip is at a master station or at a slave station.

In Figure 5, blocks 50 and 51 correspond to Figure 1, above, blocks 52 and 53 correspond to Figure 2, blocks 54 to 60 correspond to Figure 3, while block 61 corresponds to Figure 4.

The input OER to blocks 50 and 51 comes from the pseudo-random scrambler, and the input CS is a control input from a microprocessor (see Figure 6, below). The inputs on the load CS control the input of data over the Data Bus, and the handling (as described with reference to Figure 1) of the 9K6 and 64Kb/s streams.

The block 54, PLL, embodies the portions of Figure 3 which form the phase locked loop, and its inputs are 7M68, from a 7.68 Megahertz clock, and the receive clock inpout RxCLK. It has outputs to the 80K phase control and the data recovery block 58. This has an input RXD from the line interface, as will be seen in Figure 6.

The descrambler block 59 has an output from its PRS portion to the timing portion of the block 52 to maintain correct timing inter-relationships. The Serial Interface block 53, has, as shown various data inputs and outputs.

One use for two of the chips is shown in Figure 6, where the left hand assembly, i.e. the Master, with parallel input/output, is at a telephone exchange while the right hand assembly, i.e. the Slave, with serial input/output, is at a subscriber's terminal. However, it will be appreciated that the chip has applications to other systems where digital links are used.

The microprocessor 70 at the Master end has inputs and outputs, a signalling bus, and has an input from a 64K – 9K6 bus, which also feeds the chip 71. This bus conveys signals which indicate what data has to be dealt with, while that data enters via the parallel bus SW0–SW7. The connection from the chip 71 to the line interface 72 and single, including receive clock RxCLK, data for transmission PTxD, received data RxD and receive Control RxD Control.

At the Slave end the chip 75 is coupled to another microprocessor 76, data interface 77 for data sent on the 9K6 channel and a CODEC 78 for dealing with voice. The block 77 has 9K6 receive and transmit connections 9K6RxD and 9K6TxD, and a clock input with two clocks, one at 9K6, i.e. 9.6 KHz and one at 960KHz. The Codec 78 has 64RxD, receive, and 64KTxD, transmit, inputs, with a clock input 80KCLK, because the speech still includes 6K4 bits, and a 64K Sync input.

Microprocessor-chip connections at the two ends are similar, i.e. MD0–MD3 for transfer of signalling information from the 6K4 channel between the chip and the microprocess. Other connections are Microprocessor Read MREAD, MHIGH, MSIGNAL, MDISABLE and INT, used for various interworkings between microprocessor and chip.

CLAIMS

1. An electronic circuit unit for data handling, the data being expressed in a diphase code, wherein one binary value is represented by 01 and the other binary value by 10, so that 01 and 10 are correct bit conditions and 00 and 11 are incorrect bit conditions, in which if *a* such incorrect bit conditions are counted by a first counter before or at the same time as *b* such correct conditions are counted by a second counter it is assumed that an error condition exists, in which outputs from the counters cause inversion of the phase of the outputs from a clock circuit used in association with the data recovery means when such an error condition is detected, the output of such data being interrupted when such an error condition is detected, in which if the second counter counts *b* successive correct bit conditions before the first counter counts *a* incorrect conditions the data output occurs, and in which on each occasion that the counters detect an error and also each occasion that the second counter reaches *b* the counters are cleared.

2. An electronic circuit unit as claimed in claim 1, in which $a = 2$ and $b = 16$, in which data bits as derived from the data recovery means are assembled into blocks and emitted from the unit in block-wise manner, under control of the counters.

3. An electronic circuit unit as claimed in claim 1 or 2, and which forms part of an interface between a telephone exchange or a subscriber's terminal and a line.

4. An electronic circuit unit for data handling, substantially as described with reference to the accompanying drawings.

Printed for Her Majesty's Stationery Office by Croydon Printing Company (UK) Ltd, 2/87, D8817356.
Published by The Patent Office, 25 Southampton Buildings, London WC2A 1AY, from which copies may be obtained.